

# PostCSS

— HOW TO INSTALL —



**IN A ROCKET**

*Learn front-end development at rocket speed*

# INSTALL PREREQUISITES

*What do you need before installing PostCSS*

# 1 NODEJS: DOWNLOAD & INSTALL

## 1. DOWNLOAD

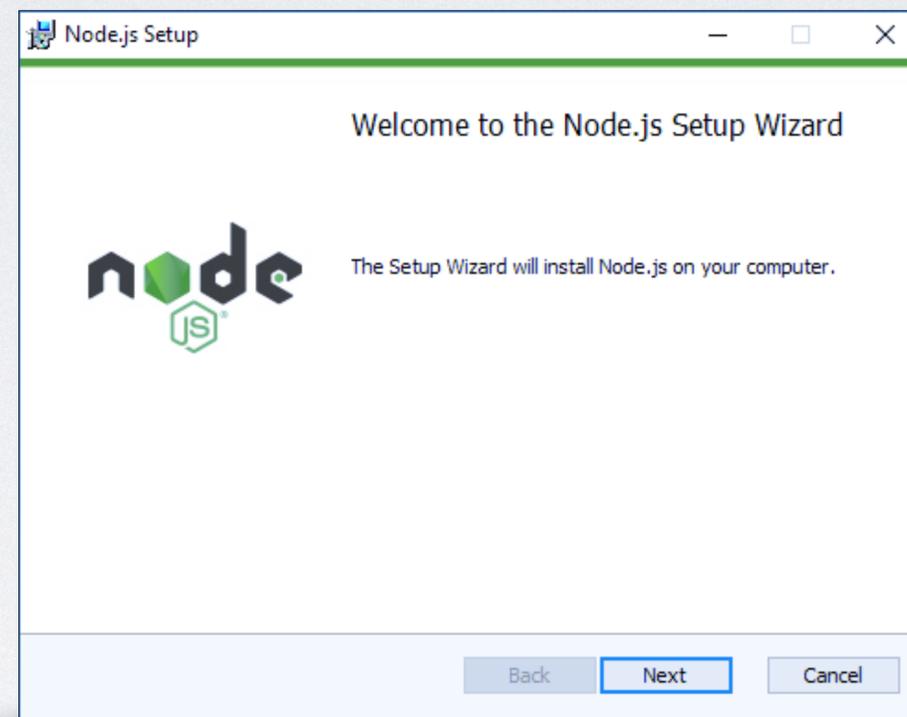
Download the right version for your platform.

[nodejs.org](https://nodejs.org)



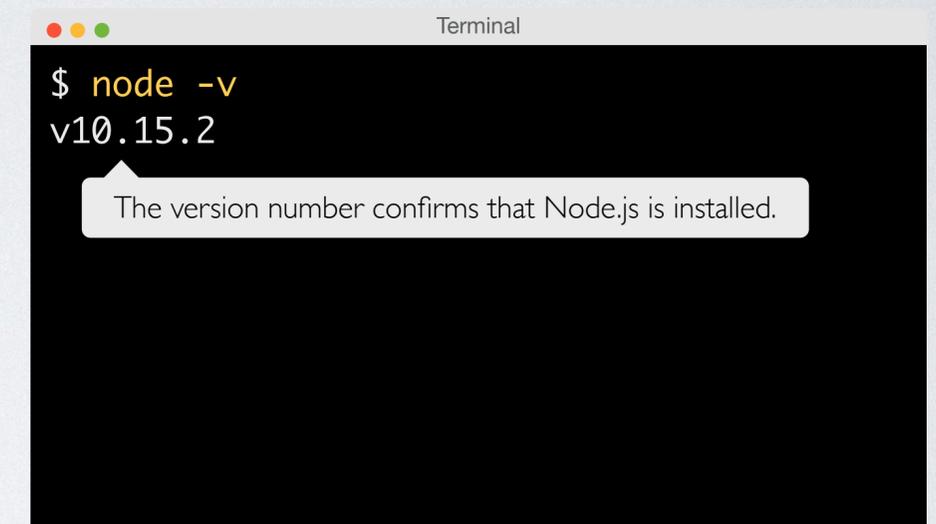
## 2. INSTALL

Accept all defaults when installing.



## 3. CHECK

Check that nodejs has been installed correctly.



To enter this command (node -v) you can use:

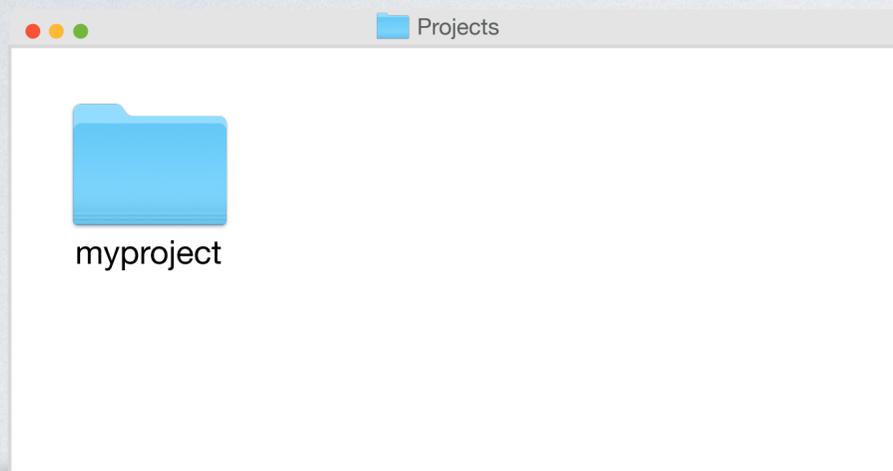
🍏 Terminal application included with MacOS.

🪟 Node.js Command Prompt (available after installing Node.js).

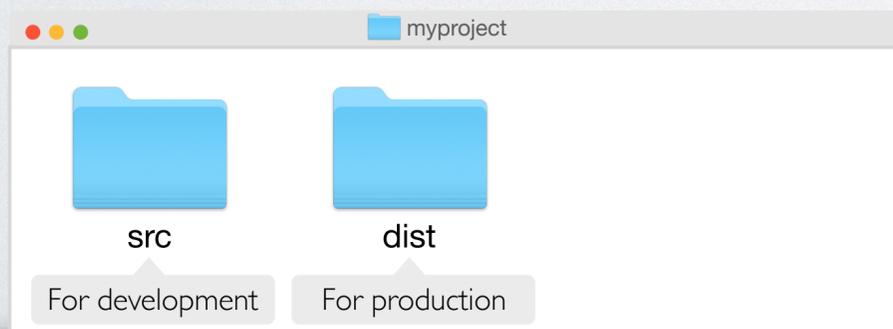
# 2 PROJECT FOLDERS & PACKAGE.JSON FILE

## 1. PROJECT FOLDERS

Create a folder for your web site / application.

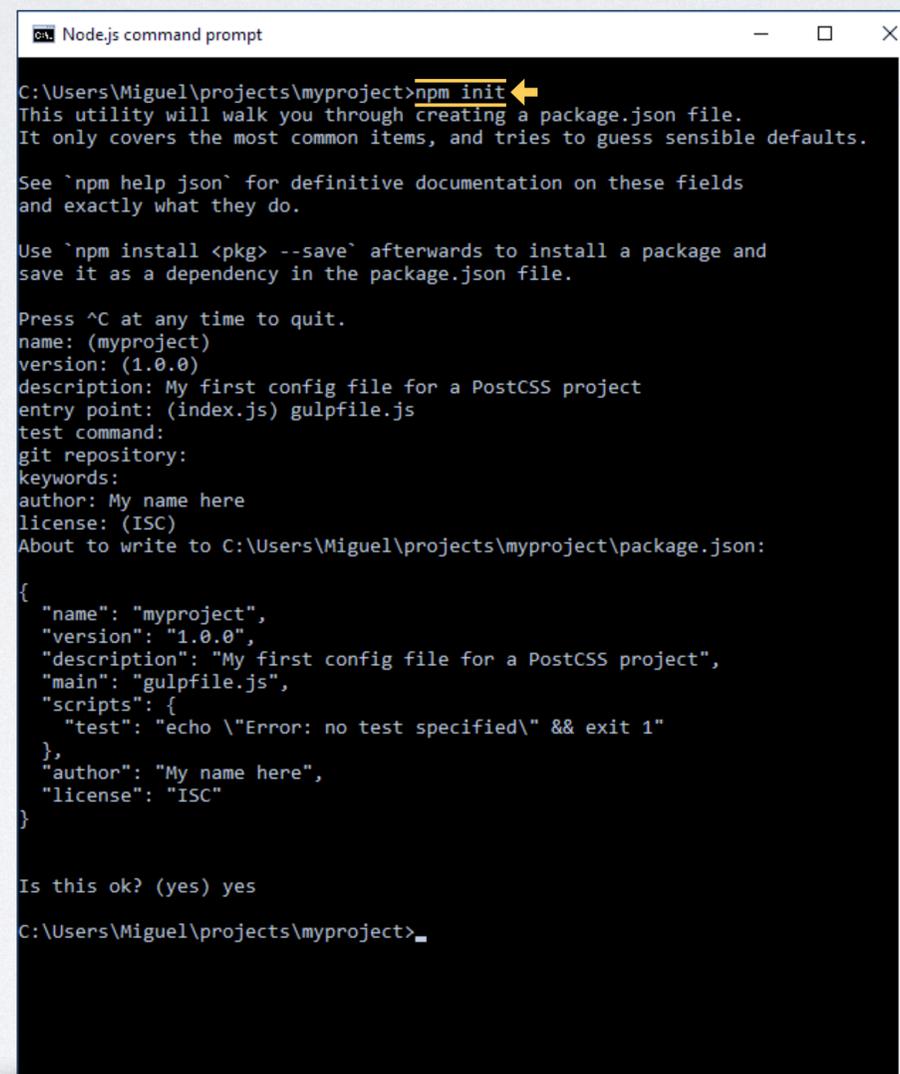


Create two subfolder inside for your source and distribution code.



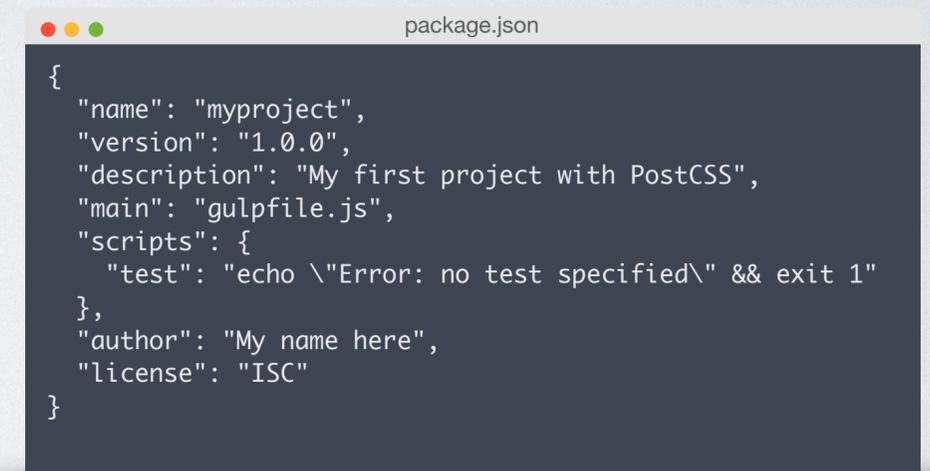
## 2. PACKAGE.JSON

Run `npm init -y` in that folder to create the package.json file.



## 3. CHECK

Open the package.json file that `npm init` have just created.



You should find this file in the root of your project folder.

# 3 GULP: INSTALL

## 1. GLOBALLY

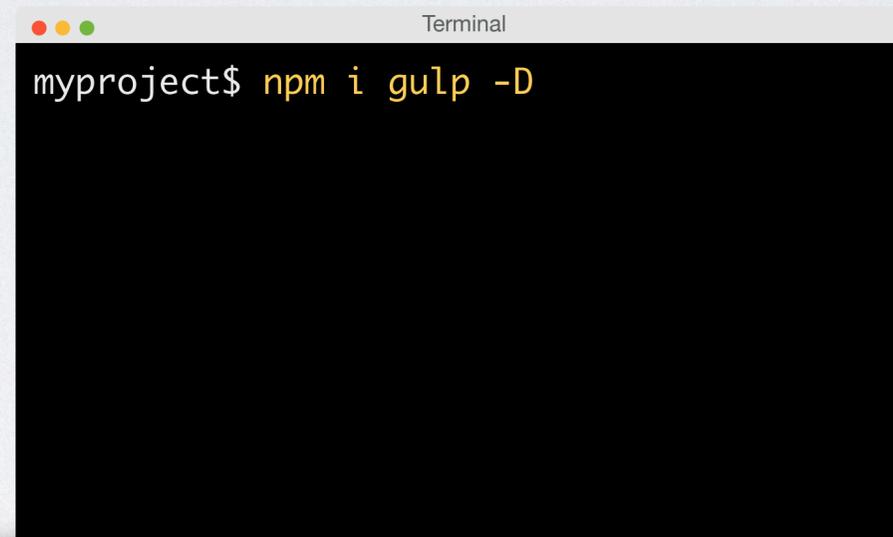
Install Gulp globally.



```
Terminal  
myproject$ npm i gulp -g
```

## 2. FOR YOUR PROJECT

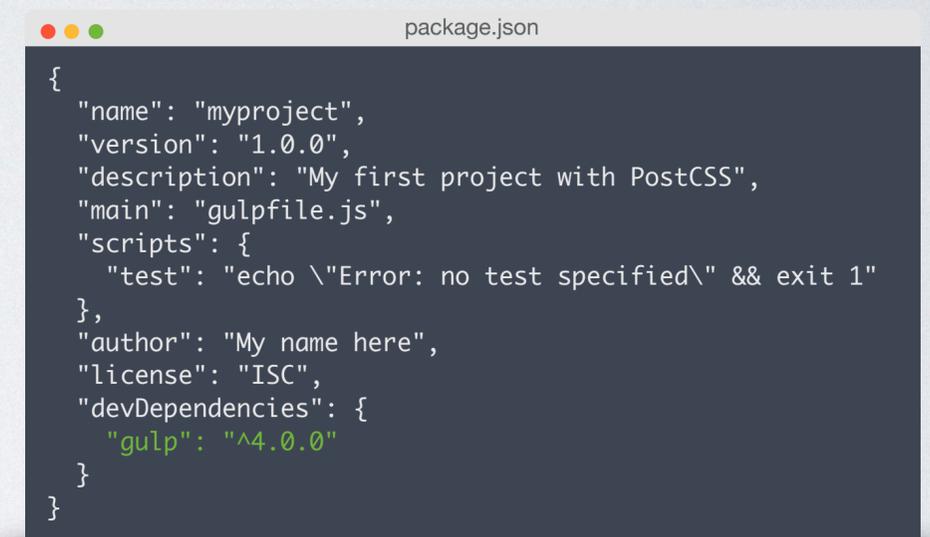
After installing Gulp globally, we need to install it for our project.



```
Terminal  
myproject$ npm i gulp -D
```

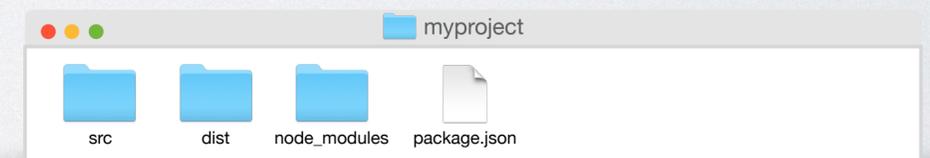
## 3. CHECK

Gulp should now appear in the package.json file.



```
package.json  
{  
  "name": "myproject",  
  "version": "1.0.0",  
  "description": "My first project with PostCSS",  
  "main": "gulpfile.js",  
  "scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1"  
  },  
  "author": "My name here",  
  "license": "ISC",  
  "devDependencies": {  
    "gulp": "^4.0.0"  
  }  
}
```

It's included as a developer dependency.



A folder named node\_modules should also appear in the root of your project folder.

 PostCSS needs to be integrated into your workflow through a build tool. We're going to use Gulp as the best option for this course. But you can use PostCSS with other popular build tools. [More information.](#)

# INSTALL POSTCSS

*How to easily complete the installation*

# POSTCSS: INSTALL

## INSTALL

After completing the prerequisites, you're ready to install PostCSS.

```
Terminal
myproject$ npm i gulp-postcss -D
```

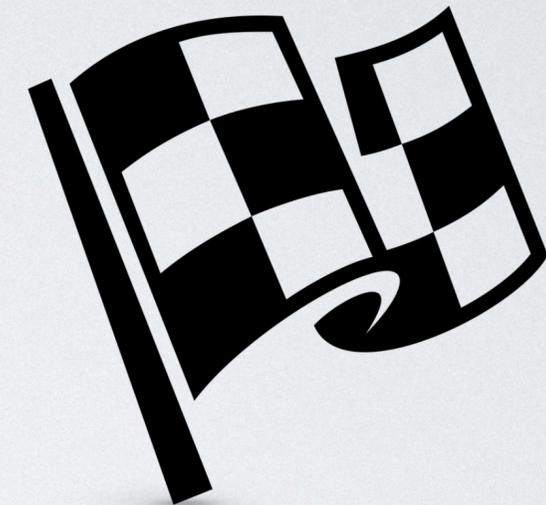
## CHECK

gulp-postcss is added as a dependency to your package.json.

```
package.json
{
  "name": "myproject",
  "version": "1.0.0",
  "description": "My first project with PostCSS",
  "main": "gulpfile.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "My name here",
  "license": "ISC",
  "devDependencies": {
    "gulp": "^4.0.0",
    "gulp-postcss": "^8.0.0"
  }
}
```

## LET'S GO!

Everything is OK to get started. Let's have some fun with PostCSS!



# FIRST RUN

*Running PostCSS before installing any plug-in*

# 1 POSTCSS: FIRST RUN

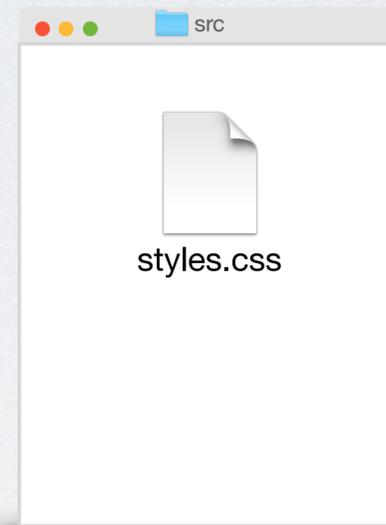
## CREATE YOUR CSS

Code your original stylesheet.

```
styles.css
::placeholder { color: gray; }
a {
  -webkit-border-radius: 5px;
  border-radius: 5px;
}
```

## SAVE TO SRC

Save your CSS into the src folder.



# 2 POSTCSS: FIRST RUN

## PREPARE YOUR TASK

After installing Gulp, create `gulpfile.js` with the following code.

```
gulpfile.js
var gulp = require('gulp'),
    postcss = require('gulp-postcss');

gulp.task('css', function () {
  var processors = [
    // PostCSS plugins here
  ];
  return gulp.src('./src/*.css')
    .pipe(postcss(processors))
    .pipe(gulp.dest('./dist'));
});
```

Save this file in your `myproject` folder.

READY TO USE CODE

## RUN

Run the task you've just created.

```
Terminal
myproject$ gulp css
```

## CHECK

Compare the original and the generated files.



Both files are identical. PostCSS has done its work but, as we haven't added any plugin yet, the original and the generated CSS have the same code.

# PostCSS

— HOW TO INSTALL —



**IN A ROCKET**

*Learn front-end development at rocket speed*